DISEÑO Y CONSTRUCCIÓN DE UN ROBOT ARTICULADO QUE EMULA EL MOVIMIENTO DE UN GUSANO

Cruz Dávalos Patricio J. , Ing.
Villegas Santillán Cristóbal R. ,Ing.
Sotomayor Orozco Nelson G., MSc.
Escuela Politécnica Nacional

Resumen
Los robots sin patas son conocidos como *robots gusano o robots serpiente*. Estos no son tan veloces ni tan ágiles como los que tienen patas, pero tienen una serie de ventajas que justifican su investigación.
El objetivo principal es el diseño y construcción de un robot tipo gusano que se desplace horizontalmente y en línea recta de forma análoga a como lo hacen los gusanos, mediante ondas transversales que se propagan desde la cola hasta la cabeza (secuencia de elevación) y que pueda realizar giros (secuencia de orientación).
Para cumplir con este objetivo se realizó un enlace entre el computador y el hardware del gusano robot, utilizando comunicación serial. Además se diseñó un HMI (Human Machine Interface) utilizando Microsoft Visual Basic 6.0 el cual es amigable de manera que el usuario pueda manipularlo con facilidad.

Abstract
Robots without legs are known as *worm robots* or *snake robots*. They are not as fast or as agile as the ones with legs, but they have some advantages that justify their research. This has led to consider the present project which has as main objective to design and build a *worm robot* prototype that moves horizontal or vertically, in straight lines as worms do. They are moved by transversal waves that are spread from tail to head (elevation sequence), and also, they can turn (sequence of direction).
In order to comply with that objective, a link between the computer and the hardware of the *worm robot* will be made by using serial communication. Besides an HIM (Human Interface Machine) program has been designed using Microsoft Visual Basic 6.0, which is friendly so the user can handle it easily.

1. INTRODUCCIÓN

Hoy en día la robótica como disciplina ha crecido significativamente y existe una gran cantidad de robots, cada uno de estilo totalmente diferente al otro, pero todos utilizando la base fundamental de la electrónica.

Debido a que el hombre constantemente busca medios y métodos para facilitar su investigación, se han creado los *robots de investigación,* con estos se intenta aportar algo nuevo a la robótica: nuevos algoritmos, nuevas formas de movimiento, etc. Dentro de estos, y con la idea de buscar robots que logren sustituir a los vehículos con ruedas de manera que puedan moverse por superficies donde estos no pueden, ingenieros y científicos están imitando animales, que se encuentran bien adaptados al medio en el que viven. Una clasificación de estos es: los robots con patas y los "sin patas".

Los robots sin patas son conocidos como *robots gusano o robots serpiente*. Al tener una forma alargada, pueden penetrar por sitios a los que otro tipo de robot no tiene acceso, como por ejemplo una tubería. Pueden adoptar la forma de la superficie por la que se desplazan, de manera que pueden hacerlo por sitios muy tortuosos. Y están compuestos por segmentos iguales, que se enganchan unos a otros pudiendo conseguirse robots de cualquier longitud.

2. MECANISMOS DE MOVIMIENTO [1]

El desplazamiento se consigue mediante ondas que recorren el robot, para lo cual las articulaciones deben estar perfectamente coordinadas. Estas ondas pueden ser:

Ondas Transversales.- en las que el movimiento de la onda es perpendicular a la dirección de avance, según como esté situado el plano que contiene a la onda con respecto al plano por el que se desplaza el robot se clasifican en:

Ondas Transversales Paralelas.- es el clásico movimiento de las serpientes haciendo "eses". También se denomina movimiento de serpentina. Las ondas se transmiten desde la cabeza hasta la cola generando movimiento.

Ondas Transversales Perpendiculares.- es el movimiento de los gusanos de seda. Aquí las ondas se transmiten desde la cola hasta la cabeza.

Ondas Longitudinales.- el movimiento de la onda se realiza en la misma dirección de propagación. Las ondas son como las del sonido al propagarse; contracción–expansión–contracción. Es el movimiento de las lombrices de tierra.

2.1 Gusano Longitudinal

Los gusanos longitudinales se definen como aquellos que son recorridos por ondas

longitudinales, desde la cola hasta la cabeza. Un ejemplo podrían ser las lombrices de tierra.

Se supone que el gusano está constituido por N segmentos iguales, de longitud L_0 cuando están en reposo. Los segmentos son contráctiles y retráctiles de manera que se pueden contraer y expandir una distancia máxima. Todos los segmentos son exactamente iguales, con la misma masa y propiedades mecánicas y están unidos unos a otros sólidamente, no pudiendo separarse.

El movimiento del gusano es una evolución desde un estado inicial hasta otro final. Cada instante de tiempo define un estado del gusano. Esta evolución puede ser continua, en ese caso se habla de *movimiento continuo*, pero puede ser discreta si los vectores de estado no evolucionan continuamente, sino que lo hacen discretamente. En ese caso el movimiento entre el estado inicial y final queda determinado por una sucesión finita de estados: *secuencias*.

El estudio teórico de estos gusanos es muy interesante y permite presentar conceptos que serán de gran ayuda para un Gusano Transversal. Sin embargo la implementación, es decir, la construcción de uno de ellos, es muy complicada por varios motivos:

1) El gusano se "arrastra" por lo que hay mucho rozamiento con la superficie. Para eliminar este rozamiento en las partes contráctiles habría que elevarlas del suelo, lo cual complica mucho la mecánica.
2) Hay que sincronizar muy bien todos los mecanismos. Un punto debe estar fijado sólo en unos instantes de tiempo determinados.

2.2 Gusano Transversal

Se define como gusano transversal, a aquel que se mueve mediante ondas transversales que recorren su cuerpo desde la cola hasta la cabeza. Es el movimiento normal de muchos tipos de gusanos, como por ejemplo los de seda.

Los gusanos transversales se modelan mediante dos elementos: las articulaciones y los segmentos. Y están constituidos por *N* articulaciones, denominadas: $a_1, a_2, \ldots, a_N$; y, *N-1* segmentos: $S_1, S_2, \ldots, S_{N-1}$.

Los segmentos son rígidos, de longitud *L* y todos ellos exactamente iguales, con la misma masa y las mismas propiedades mecánicas. El segmento de cola es el S_1 y el de cabeza el S_{N-1}.

Las articulaciones son los elementos que unen dos segmentos entre sí. En total hay *N*, de las cuales dos son ficticias, a_1 y a_N puesto que no unen dos segmentos y en la implementación no se corresponden con ninguna articulación física. Las restantes, *N-2*, son reales. La articulación a_1 se denomina cola (*articulación de cola*) y la a_N, cabeza (*articulación de cabeza*). Los segmentos se encuentran perfectamente enganchados unos a otros y el estado en el que se encuentra cada articulación sólo depende del estado interno del gusano y no viene impuesto por fuerzas externas.

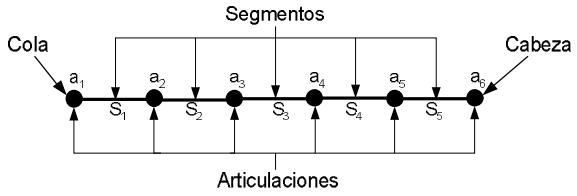

Figura 1 Modelo del gusano transversal

Las articulaciones se caracterizan por el ángulo de estado $\varphi_i(t)$, que determina el ángulo entre el vector interior y exterior de la articulación i (Figura 2). Para caracterizar el estado en el que se encuentra el gusano se utiliza un vector de estado *E*, de *N* componentes, una para cada articulación. La componente i de este vector es el ángulo de estado de la articulación i, en un instante t:

$$E(t) = (\varphi_1(t), \varphi_2(t), \ldots, \varphi_N(t))$$

Las articulaciones a_1 y a_N son ficticias por lo que su valor carece de importancia para la implementación, sin embargo son muy útiles para los cálculos.

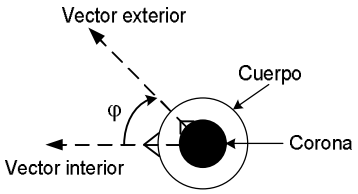

Figura 2 Elementos de una articulación

2.2.1 Función de Contorno

La función de contorno es la herramienta empleada para lograr que el gusano evolucione, pasando de un estado a otro. Se trata de una función *f:* R→R que indica la relación entre las coordenadas (x, y) de las articulaciones. La función de contorno determina los ángulos de las articulaciones, pero no de una manera directa.

Una función de contorno define el estado del gusano transversal para un cierto instante t. Si esta función se propaga con una velocidad *v*, se generan los diferentes vectores de estado en función del tiempo. Con ello se consigue una evolución.

Para que exista avance en todo instante de tiempo al menos hay un punto del gusano que nunca retrocede, por lo que el avance es posible.

Además, la evolución sólo será totalmente correcta si el gusano es continuo. En los discretos siempre será incorrecta, pero se acercará más a la evolución correcta cuanto más se parezca la forma que adopta el gusano a la función de contorno.

En la implementación de un gusano las distancias entre articulaciones (L) no se pueden variar y hay un número fijo de ellas. Por tanto, para conseguir que el movimiento sea más perfecto, habrá que jugar con la función de contorno.

Los gusanos transversales son más sencillos de implementar que los longitudinales, debido a que el movimiento no es de deslizamiento. Por ello depende menos de la superficie por la que se mueven, puede ser lisa, rugosa, con pequeños obstáculos, etc. La regla de avance sólo se cumple al 100% en el caso de los gusanos continuos. En una implementación, lógicamente, este modelo no es posible y tiene que ser discreto. El diseñador debe elegir con cuidado las funciones de contorno.
Las articulaciones se pueden implementar mediante servomecanismos, que se encuentran fácilmente en el mercado y con diferentes características mecánicas.

2.3 Algoritmo de Ajuste
Para mover un gusano transversal, virtual o real, se necesita generar una secuencia de vectores de estado E(t), que definan la evolución interna. Una vez obtenidos, se los puede almacenar y cargar en aplicaciones que muevan gusanos virtuales o reales.

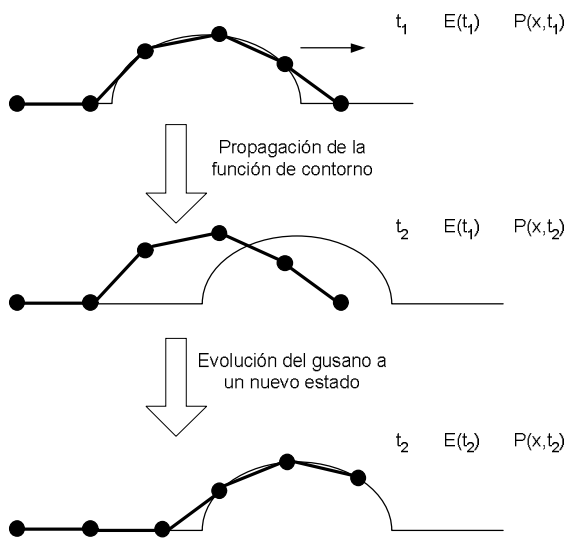

Figura 3 Bases para la generación de una secuencia

El problema que se quiere resolver es el problema de ajuste (Figura 3). Una vez conocida la solución al problema de ajuste, es muy sencillo obtener una secuencia. Sólo hay que partir de un estado inicial conocido, desplazar la función de contorno, aplicar la solución para obtener el siguiente estado y repetir el proceso hasta alcanzar nuevamente el estado inicial. Esta secuencia caracteriza ese movimiento y se puede hacer que el gusano la repita cíclicamente, consiguiéndose el movimiento de avance deseado. Y si se conoce la solución para una articulación, se puede ajustar el gusano entero, aplicando una iteración desde la articulación 1 hasta la N.
Suponiendo una situación genérica mostrada en la Figura 4, lo que se quiere encontrar son las coordenadas (x, y) del punto de intersección entre la Función de Contorno y la circunferencia con centro en la articulación 1 y radio L, que pasa por la articulación 2.

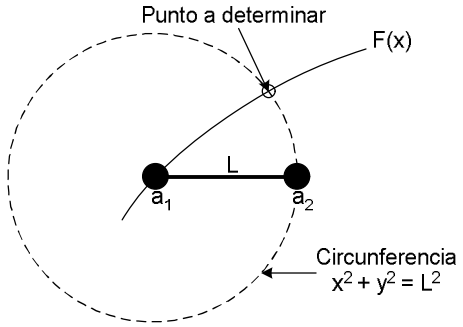

Figura 4 Cálculo del punto final

De la Figura 4 se obtiene la ecuación de ajuste:

$$x^2 + F(x)^2 = L^2$$

que permite calcular el valor de *x* del punto ajustado. La solución numérica consiste en dar valores a *x* hasta que el error sea lo más cercano posible a 0. La función de error empleada es:

$$e(x) = \left| x^2 + [F(x)]^2 - L^2 \right|$$

Esta función toma el valor 0 cuando el valor de *x* cumple la ecuación de ajuste. Cuanto mayor sea el error, más alejado está el valor *x* del valor correcto.
Es más intuitivo resolver el problema de una manera gráfica relacionada con el problema de ajuste, en lugar de resolver una ecuación. A este procedimiento se le denomina *algoritmo de ajuste*.
El algoritmo en cuestión es el siguiente. Se quiere que la articulación *i* se sitúe sobre la función de contorno, realizando una rotación de la articulación *i-1*, que ya se encuentra sobre la función de contorno. Lo que se hace es rotar la articulación *i*, con respecto a la *i-1* un ángulo $\Delta\theta$ y se va evaluando la función de error hasta encontrar un valor mínimo. Momento en el cual ya está ajustada la articulación. Cuando menor sea el incremento angular, mayor será la precisión en el ajuste, pero más tiempo se tardará en ajustar.
Para aplicar el algoritmo correctamente, hay que saber a priori hacia qué dirección hay que rotar la articulación para que el error disminuya. Eso se calcula fácilmente mediante la siguiente función:

$$S = F(x) - y$$

Si $S > 0$ hay que incrementar el ángulo θ, de manera que la articulación se aproxime a la función

de contorno por abajo y si $S < 0$ hay que decrementarlo y entonces la aproximación es por arriba.

Se puede partir de un ángulo inicial de ataque. En la Figura 5 se indica un ejemplo. La articulación a_i se encuentra en una posición inicial (x, y) conocida. Teniendo en cuenta el triángulo formado por los puntos *B*, a_{i-1} y a_i, un *ángulo inicial de ataque* óptimo es aquel que está mucho más cercano al ángulo final de ajuste, y es:

$$\tan(\theta) = \frac{F(x)}{x} \Rightarrow \theta = \arctan\left[\frac{F(x)}{x}\right]$$

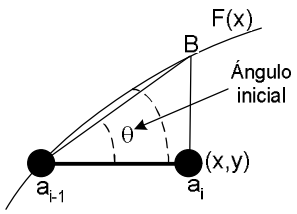

Figura 5 Ángulo inicial de ataque

2.4 Gusano Plano Rígido

Es aquel que tiene *N* articulaciones y *N-1* segmentos rígidos, de longitud L_o, que está situado sobre el plano *xy*. Las articulaciones se mueven de manera que el gusano siempre permanece dentro del plano. El modelo es similar al de un gusano transversal, con la diferencia de que está en el plano *xy*. Las articulaciones están colocadas de manera que los segmentos interior y exterior se mueven permaneciendo siempre en el plano *xy*

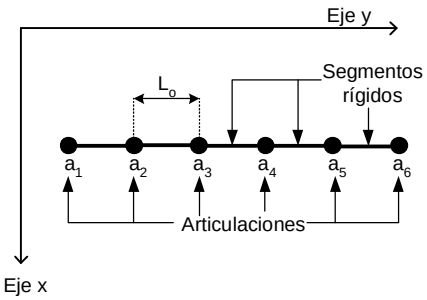

Figura 6 Gusano plano rígido

Cada articulación queda caracterizada por su ángulo de orientación $\theta_i(t)$ y el gusano completo por el vector de orientación, $E_o(t)$, que tiene *N* componentes que indican los estados de orientación de cada articulación:

$$E_o(t) = (\theta_1(t), \ldots, \theta_N(t))$$

Las articulaciones a_1 y a_N, cabeza y cola, son ficticias y no se implementan en el gusano.

2.4.1 Función de Trayectoria

La función de trayectoria indica las posiciones dentro del plano *xy* que deben cumplir todas las articulaciones del gusano. Es decir, el gusano debe estar ajustado a la función de trayectoria en todo momento. La forma de estas trayectorias puede ser cualquiera, no obstante se ha trabajado con trayectorias simples, constituidas por dos segmentos que forman un determinado ángulo.

Claramente se observa que se puede expresar la evolución interna mediante una función de trayectoria que se propaga desde la cabeza hasta la cola, y a la que el gusano se va ajustando.

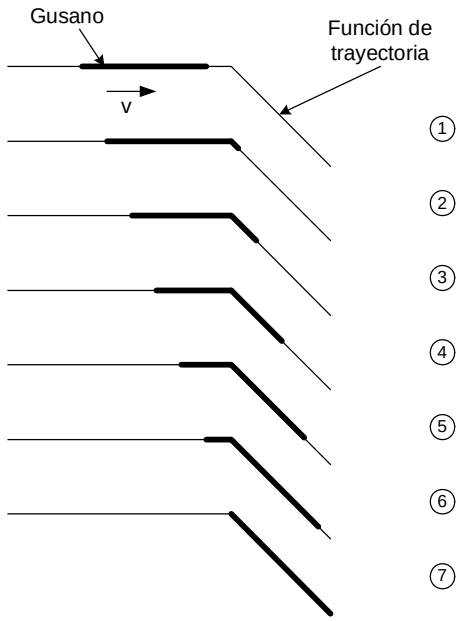

Figura 7 Gusano avanzando por una Función de Trayectoria

Para calcular los estados internos se hace exactamente igual que en el caso del gusano transversal. Se aplica el algoritmo de ajuste con respecto a la función de trayectoria, empezando por la articulación de la cola, donde está situado el origen y avanzando hasta tener el gusano completamente ajustado. El siguiente estado se obtiene desplazando hacia la izquierda la función de trayectoria y volviendo a aplicar el algoritmo de ajuste.

2.5 Gusano Tridimensional

Se entiende por gusano tridimensional, aquel constituido por *N* articulaciones de doble estado y *N-1* segmentos rígidos. Estas articulaciones permiten orientar un segmento en cualquier dirección del espacio. De esta forma, el gusano tridimensional puede tener partes que estén fuera del plano *xy*, a diferencia de los gusanos planos.

2.5.1 Articulaciones de doble estado

Son articulaciones que tienen dos grados de libertad, todo ello en un único segmento. Son articulaciones ficticias, no existen en la realidad, pero son muy útiles para modelar. Se suponen como “nodos” con dos segmentos conectados, los cuales se pueden orientar en cualquier dirección del espacio.

En la Figura 8 se ha graficado una de estas articulaciones con un segmento. Los ángulos que definen su estado son: φ, *ángulo de elevación* y θ, *ángulo de orientación.*

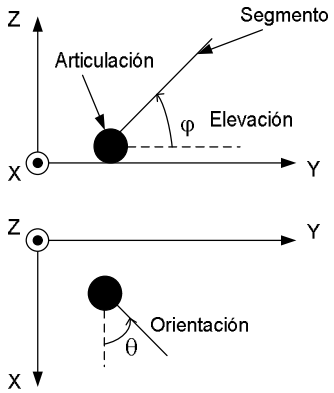

Figura 8 Articulación de doble estado

Las articulaciones se caracterizan por el ángulo de orientación $\theta_i(t)$ y el de elevación $\varphi_i(t)$, necesitando entonces dos vectores de *N* componentes para caracterizar el gusano tridimensional:

- *Vector de orientación:*
$$E_o(t) = (\theta_1(t), \ldots, \theta_N(t))$$
- *Vector de elevación:*
$$E_e(t) = (\varphi_1(t), \ldots, \varphi_{N-1}(t))$$

Estos dos vectores forman el *vector de estado del gusano tridimensional:*

$$E(t) = [E_e(t), E_o(t)]$$

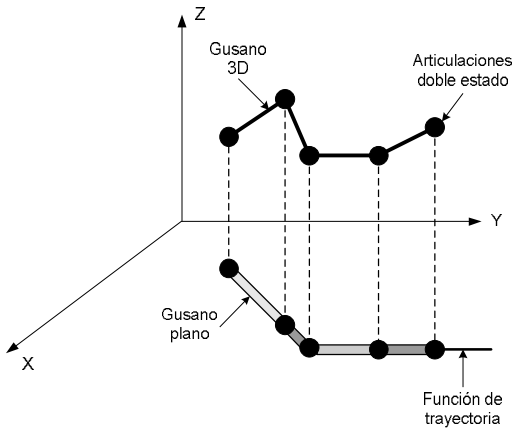

Figura 9 Gusano tridimensional y su proyección

La proyección del gusano tridimensional sobre el plano *xy*, es un gusano plano, en el cual la longitud de los segmentos de dicho gusano depende de los estados de elevación. El estado de orientación del gusano tridimensional y del gusano plano es el mismo en todo momento, por lo que es más sencillo trabajar con la proyección para determinar estos estados (Figura 9).

2.6 Algoritmo de Giro

El gusano tridimensional está caracterizado por su estado interno, constituido por los vectores de elevación y por los vectores de orientación, por lo que para conseguir el movimiento es necesario coordinar estos estados.
Existen dos entradas muy importantes: *la función de contorno*, que determina la evolución de los estados de elevación; y, *la función de trayectoria* que determina los estados de orientación. Para obtener el estado final hay que coordinar estas dos evoluciones.

El problema que presentan estos gusanos es el de la articulación de doble estado. Este tipo de articulación no se encuentra en el mercado y se debe implementar a partir de dos servos, uno para la orientación y otro para la elevación. Pero el control mediante propagación de ondas es viable pudiendo conseguir que además de avanzar el gusano realice giros. Es por esto que, teniendo en cuenta el algoritmo de giro e implementando una adecuada articulación de doble estado, se puede lograr implementar un gusano tridimensional.

3. DISEÑO Y CONTRUCCIÓN DEL PROTOTIPO

3.1 Módulo Y1

Se ha buscado un posible diseño mecánico para los módulos que constituirán el robot gusano, al cual se ha decidido denominar Blender. De esta forma, se ha encontrado un módulo (Y1), que se ajusta a los requerimientos de este proyecto, el cual es realizado por Juan González Gómez en su tesis doctoral “Diseño de Robots Ápodos” [2].
Los criterios de diseño de los módulos Y1 fueron los siguientes:

- Sencillez, tienen que ser lo más sencillos y simples posibles, con el menor número de piezas para que se puedan construir prototipos a mano.
- Fabricables, deben poder ser fabricados industrialmente por lo que se realizan planos detallados.
- Conexión en fase y en desfase, esta es la idea tomado de los módulos G1 de Polybot. [3]
- Facilitar cableado, que los cables puedan pasar de un módulo a otro.
- Abiertos, ninguna restricción impuesta en su fabricación/utilización, para que cualquiera los pueda usar.

Es por esta última característica por la que los planos de los módulos se pueden utilizar libremente [4].
En la Figura 10 se muestra el módulo en el que la cabeza está situada en dos ángulos diferentes con respecto al cuerpo,

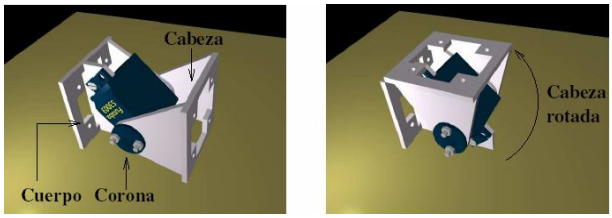

Figura 10 Módulo Y1

Una característica fundamental es la de poder colocar dos módulos en fase o desfasados, como se muestra en la Figura 11, esto permite construir gusanos que se puedan mover en un plano (conexiones desfasadas), o bien un gusano que sólo avance en línea recta (todos los módulos en fase).

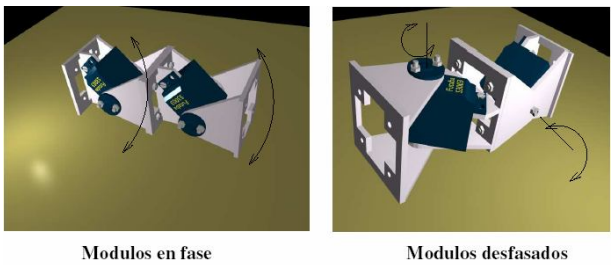

Figura 11 Conexión de dos módulos Y1

Las bases de los módulos tienen un hueco en el interior, lo que resulta útil para pasar los cables de un módulo a otro.

3.2 Estructura Mecánica del Prototipo

En base a los Módulos Y1 se realizó la construcción del prototipo del gusano robot Blender.

Se utilizan, estos módulos por su característica de permitir la conexión en fase y desfase, ya que en el presente proyecto se necesita el desfase para poder efectuar el movimiento de Blender en un plano y que sea capaz de cambiar de dirección, es decir pueda efectuar giros (Figura 12).

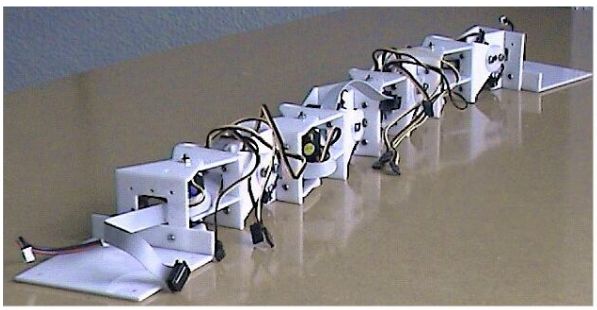

Figura 12 Estructura mecánica del prototipo

Pero se han realizado algunos cambios respecto al Módulo Y1 original, estos son:

1) Los servomecanismos empleados no son los Futaba 3003 sino los Hitec HS-311, cuyas características y dimensiones son similares a las del Futaba.
2) El material empleado en la fabricación de las piezas de los módulos Y1 no es PVC expandido, sino láminas de acrílico de color blanco de 3mm de grosor. El cual es un buen material para trabajar modelos a escala.
3) El número y tipo de piezas de sujeción, tornillos y tuercas, difiere del número utilizado en el módulo original.
4) Se han colocado dos nuevas piezas que serán el soporte de las tarjetas de control (Master y Slave) y de la alimentación de las mismas.
5) La denominación de las piezas no es la misma, ya que se vio la necesidad de volver a nombrarlas para aclarar la explicación del montaje.

A pesar de realizar estos cambios a los módulos construidos se los denominó módulos Y1, ya que en ellos están basados y los cambios no afectan su estructura.

3.2.1 Servomecanismos

Para mover el gusano robot se necesitan motores con control de posición, es decir que el eje se pueda situar en el ángulo que se desee. Además no es necesario que sean de revolución continua, con giro de 180º es suficiente. Un tipo de motores que cumplen las características anteriores son los servomecanismos. Los cuales son básicamente un motor de corriente directa una tarjeta controladora y un potenciómetro que mide hacia donde está ubicado en todo momento, es así como la tarjeta controladora sabe hacia donde mover al motor (Figura 13).

El control de posición lo efectúa el servo internamente mediante este potenciómetro y un control PWM interno, que se compara con la entrada PWM externa, mediante un sistema proporcional diferencial, y así modificar la posición de salida hasta que los valores se igualen, luego de lo cual el servo se detiene [5].

El tiempo en alto del pulso indica el ángulo al que debe posicionarse. Para el Hitec HS-311 se considera que en 1.5ms está en el centro y los límites recomendados por el fabricante son 0.9 a 2.1ms. La frecuencia de la señal PWM debe ser de 50Hz.

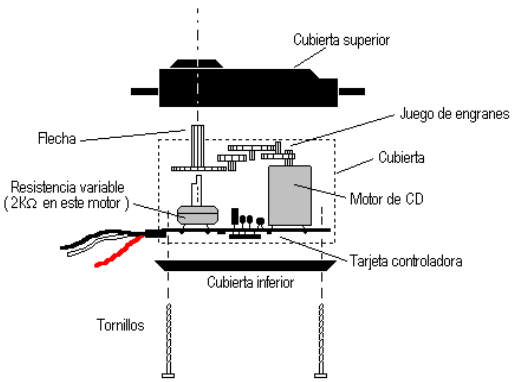

Figura 13 Piezas de un servomecanismo

3.3 Electrónica

Una vez armada la estructura mecánica del prototipo es necesario añadir la electrónica que pueda mover los servos.

El gusano tiene una serie de particularidades que hacen que la electrónica tenga que cumplir el principal requisito de ser flexible y modular, de modo que permita incorporar mejoras futuras. La solución está basada en redes de microcontroladores, la cual además debe adaptarse correctamente a la estructura mecánica del gusano.

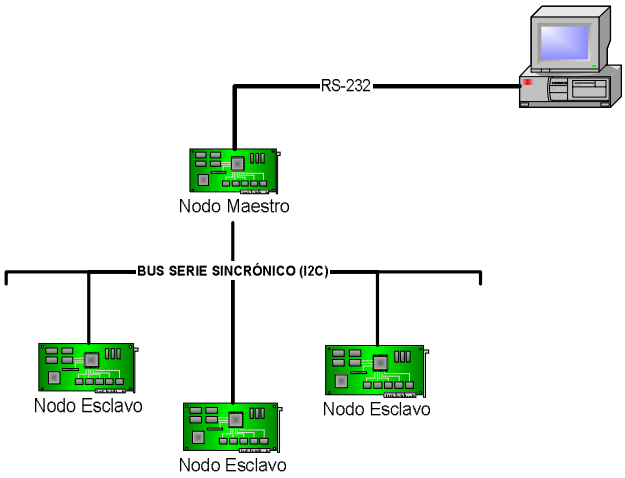

Figura 14 Arquitectura hardware

La red de microcontroladores es del tipo *maestro-esclavo* (Figura 14). En el que se tiene *el nodo maestro*, el nodo más importante, que es el que envía las órdenes a los demás nodos, *nodos esclavos*. La conexión entre los nodos es serie sincrónica, utilizando I^2C.

El nodo maestro, además es el que se conecta con la PC vía interfaz serie asincrónico, según la norma RS-232. Si se quiere trabajar en modo autónomo, se descarga la tabla de control del PC al nodo maestro para que luego se ejecute el programa actuando sobre los diferentes servos. Si se quiere un modo no autónomo, el maestro simplemente hace de puente entre el PC y el resto de nodos, pasando información de un lugar a otro. Este último se lo utiliza para calibrar los servos que conforman el gusano.

Tanto para el nodo maestro como para el nodo esclavo se utiliza un microcontrolador PIC 16F876 de Microchip. Y un nodo esclavo permite controlar hasta 8 servos del tipo Hitec HS-311.

Para la alimentación de la lógica se utilizan 4 pilas AA, mientras que la alimentación de los servos es a través de una fuente externa de 5V.

4. SOFTWARE DE CONTROL

Los programas del software de control del robot gusano Blender, involucran tanto a la red de microcontroladores del robot como también al PC, dichos programas están relacionados entre si ya que habrá un flujo de información de la Pc al microcontrolador master y de este último al slave (Figura 15).

Como se indicó, la red de microcontroladores está constituida por un nodo maestro y uno esclavo; y en cada nodo se ejecuta un software diferente.

El nodo esclavo se encarga de la recepción de tramas y de actualizar una serie de registros, además, mediante interrupción de un Timer, genera las señales PWM de control de los servomecanismos.

El nodo maestro hace de puente entre los nodos de la red y el PC, en este caso entre el PC y el único nodo esclavo. La información recibida utilizando comunicación serie asincrónica es enviada al nodo esclavo a través del bus I^2C.

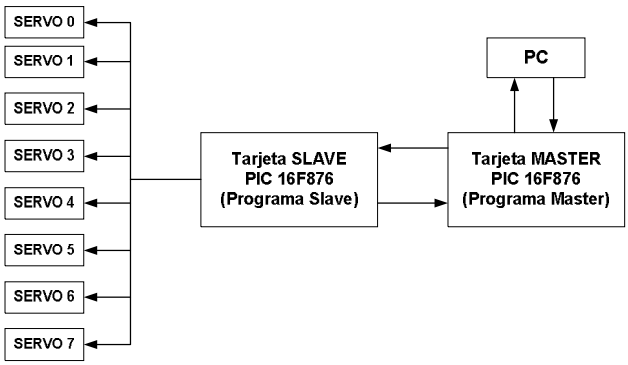

Figura 15 Arquitectura del sistema de control

Por lo cual, el control en sí es realizado por la red de microcontroladores, mientras que el cálculo de la secuencia de movimiento y la generación de la tabla de control son efectuados por la PC, la cual posteriormente es enviada al nodo maestro a través del puerto serie. Por lo que el programa del PC se subdivide en tres subprogramas, que corresponden: al cálculo de los ángulos de la Función de Contorno, al cálculo de los ángulos de la Función de Trayectoria y el cálculo-envío de la tabla de control.

La interfaz Función de Contorno, permite realizar las siguientes tareas sobre un gusano transversal virtual de 6 articulaciones y 5 segmentos:

a) Obtener el vector de estado de elevación E(t) del gusano en un instante t.
b) Cambiar los parámetros de la función de contorno: amplitud, longitud de onda, velocidad de propagación y tiempo.
c) Aplicar el algoritmo de ajuste a todo el gusano.
d) Opción de reproducción play para mover la función de contorno.
e) Opción de grabación de una secuencia en una base de datos, de manera que posteriormente pueda ser leída por la aplicación Gusano Físico.

En la Figura 16 se muestra la interfaz Función de Contorno.

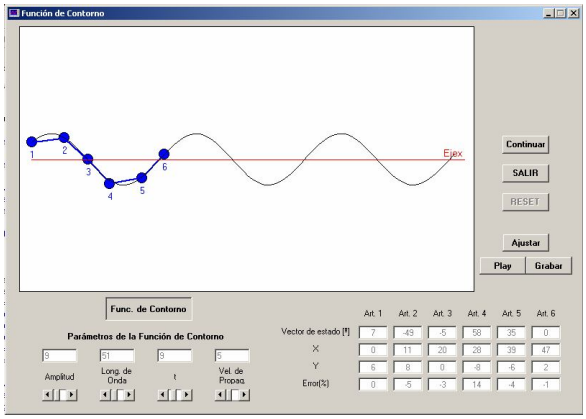

Figura 16 Interfaz Función de Contorno

La interfaz Función de Trayectoria permite realizar las siguientes tareas sobre un gusano plano virtual de 6 articulaciones y 5 segmentos:

a) Obtener un gusano plano virtual en base a los datos de elevación generados en la interfaz Función de de Contorno.
b) Obtener el vector de estado de orientación E(t) del gusano en un instante t.
c) Dibujar en pantalla la trayectoria requerida por el usuario.
d) Aplicar el algoritmo de ajuste a todo el gusano.
e) Opción de grabación de la secuencia de orientación en una base de datos, de manera que posteriormente pueda ser leída por la aplicación Gusano Físico.

En la Figura 17 se muestra la apariencia del interfaz Función de Trayectoria.

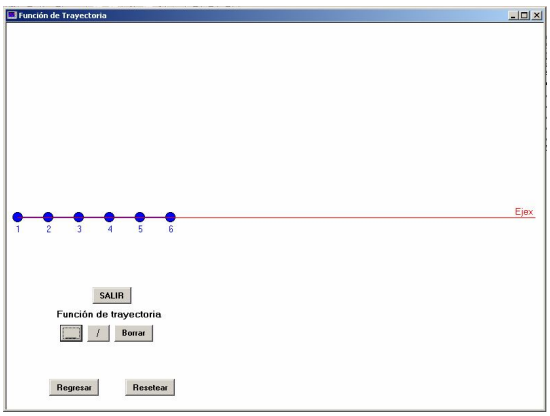

Figura 17 Interfaz Función de Trayectoria

La interfaz Gusano Físico esta diseñada para controlar el gusano real tridimensional Blender, al que se denomina gusano físico. En la Figura 18 se presenta el aspecto inicial de esta interfaz, en ella se pueden realizar las siguientes acciones:

a) Ajustar los valores de Modo y Offset de los servos independientemente
b) Mover los servos independientemente
c) Transformar los vectores de estado virtuales a vectores de estado físico.
d) Coordinar los valores de la secuencia de elevación con los de la secuencia de orientación.
e) Descargar una secuencia física a Blender.

En realidad este programa podría ser utilizado para manejar ocho servos aislados, independientemente de que se encuentren en un gusano. Además, si se cargan las secuencias generadas por el gusano virtual se consigue el movimiento del gusano físico.

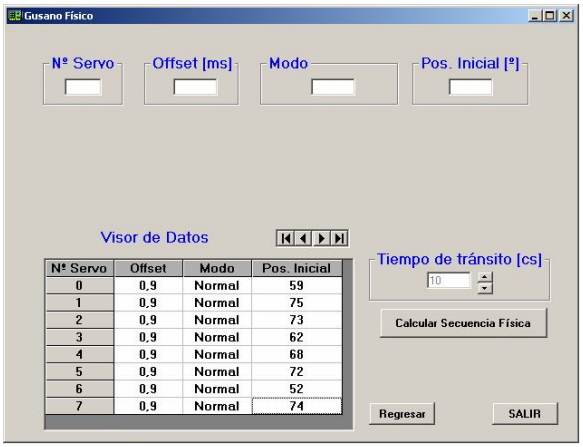

Figura 18 Interfaz Gusano Físico

5. PRUEBAS Y RESULTADOS

Una vez que se ha finalizado con la construcción del prototipo, tanto del software del PC como el robot gusano, se procedió a realizar varias pruebas para verificar el correcto funcionamiento del mismo.
Se han probado diferentes secuencias de elevación y orientación, generadas por el programa de soporte de la PC, descargadas a Blender. En todos los casos se observó que Blender las realiza sin mayores dificultades. Un movimiento adecuado se lo consiguió, por ejemplo, para una Función de Contorno con Amplitud de 7, longitud de onda de 45 y velocidad de propagación de 10, una Función de Trayectoria de un segmento de distancia 57 y ángulo de 0º, y un tiempo de tránsito de 4.
Al probar su movimiento sobre superficies lisas este no logra un avance considerable como el que se observa sobre superficies con un adecuado coeficiente de fricción. Sobre superficies con un alto rozamiento (alfombra) no tiene un movimiento totalmente libre, sus módulos no pueden rotar adecuadamente. Sin embargo, Blender emula los movimientos de un gusano.
Antes de utilizar el prototipo se deben revisar que las pilas utilizadas para la alimentación de la electrónica del robot no estén descargadas. Ya que esto ocasiona problemas al escribir y leer los datos de la secuencia en la EEPROM. Por lo que al dar la orden de iniciar el movimiento no se recuperan todos los datos de la secuencia.
La fuente de alimentación de los servos (5V y más de 1A) debe poder suplir la corriente requerida, ya

que cuando esto no sucede se observa que se forza el movimiento de los servomecanismos.
Una vez que se termina de ejecutar una secuencia por Blender, se observa que los módulos se quedan posicionados en los últimos ángulos que se hayan ejecutado. Es recomendable dar un reset al PIC Slave para que vuelva a poner a los servos en sus posiciones de default y no se forcen a los servos excesivamente.

6. CONCLUSIONES

- La relación y el análisis de los modelos de los gusanos longitudinal, transversal y tridimensional ha permitido encontrar un algoritmo de ajuste que permite calcular las secuencias de orientación y elevación de un gusano tridimensional.

- Con las ideas obtenidas del planteamiento del algoritmo de ajuste se logró diseñar un software de alto nivel para un PC que permite generar las secuencias de movimiento de elevación y orientación, en base a la manipulación de un gusano transversal virtual (secuencia de elevación) y de un gusano plano virtual (secuencia de orientación).

- Se ha construido un robot gusano, denominado Blender, cuya estructura esta constituida por 4 articulaciones de doble estado formadas por 8 módulos Y1 conectados en desfase de dos en dos (diseñados por Juan González Gómez, referencia [4]) con lo que se obtuvo un prototipo de gusano tridimensional.

- El aporte más importante en este proyecto, tanto en hardware como en software, es la implementación del desplazamiento en un plano, es decir se puedan realizar giros por parte del robot gusano.

- En el diseño mecánico del robot gusano se utilizó un material liviano pero resistente a la vez (láminas de acrílico), pero esto produjo un problema en el avance del gusano, este se resbala al realizar el movimiento sobre una superficie lisa.

- El manejo del módulo MSSP de los microcontroladores PIC 16F876 para la comunicación en la red de microcontroladores mediante la utilización del bus I2C es de fácil implementación y permite la transmisión y recepción de datos de manera eficiente.

- Una limitación del gusano es la capacidad de almacenamiento de datos en la memoria EEPROM del Master (máximo 250 valores), en la cual se guardan los datos de la secuencia enviada desde el PC. Es por esto que en ocasiones el gusano no es capaz de completar la secuencia de movimiento total planteada por el usuario.

- Es necesario actualmente enfatizar en el estudio de la robótica ya que el construir un gusano tridimensional es muy útil no sólo en el campo de los robots ápodos. En realidad se trata de un *segmento orientable* que puede tener muchas aplicaciones como por ejemplo introducir una cámara en rincones poco accesibles o incursionar en el campo de los robots modulares – autoconfigurables.

7. RECOMENDACIONES

- Al realizar la calibración de las posiciones iniciales de los servos que constituyen el robot se debe poner un cuidado especial para no ocasionar daños mecánicos a los servomecanismos o a los módulos.

- Una mejora para el prototipo construido en este proyecto sería el dotar al nodo maestro de mayor potencia de cálculo para que no haga falta disponer de un PC para el movimiento del gusano físico, o en su defecto que posea una mayor capacidad de almacenamiento.

8. REFERENCIAS BIBLIOGRÁFICAS

- [1] CUBE: El Gusano Robot (V.2.0); desarrollado por Juan González Gómez; tiene licencia GPL; http://www.iearobotics.com/personal/juan/proyectos/cube-2.0/cube.html.

- [2] CUBE RELOADED; construido y desarrollado por Juan González Gómez; tiene licencia GPL; http://www.iearobotics.com/personal/juan/doctorado/cube-reloaded/index.html.

- [3] POLYBOT, Modular Robotics; desarrollado por Mark Yim en el PARC; http://www2.parc.com/spl/projects/modrobots/ chain/polybot/index.html.

- [4] Módulos Y1; diseñados por Juan González Gómez; tiene licencia GPL; http://www.iearobotics.com/personal/juan/doctorado/Modulos-Y1/modulos-y1.html#uno.

- [5] X-ROBOTICS, robótica y micro controladores PIC, http://www.x-robotics.com/ mecanica.htm

- [6] ANGULO, J., ROMERO, S., ANGULO, I.; "Microcontroladores <<PIC>>. Diseño práctico de aplicaciones. Segunda Parte: PIC16F87X"; 2da edición; Editorial McGraw-Hill; España, 2000.

- [7] BOWLING, Stephen; "Using the PICmicro SSP for Slave I^2C Communication"; Microchip Technology Inc.; Ref. AN734; 2000.

- [8] DUQUE, Edison; "Curso avanzado de Microcontroladores PIC"; s/ed; CEKIT Compañía Editorial Tecnológica; Colombia; 1998.

- [9] FISCHER, Richard; "Using the PICmicro MSSP Module for Master I^2C Communications"; Microchip Technology Inc.; Ref. AN735; 2000.

- [10] JENSEN, C., H.; "Dibujo y Diseño de Ingeniería"; s/ed; Editorial McGraw-Hill; México; 1988.

- [11] PIC16F87X 28/40-pin 8-Bit CMOS Flash Microcontrollers; Microchip Technology Inc.; 1998.

- [12] The I^2C-BUS Specification; Version 2.1; Philips Semiconductors; Enero 2000, document order number: 9398 393 40011, http://www. semiconductors.philips.com/acrobat/literature/9398/39340011.pdf

- [13] Diseño de un módulo para control de 8 servos y 5 canales analógicos por bus I^2C; desarrollado por Alejandro Alonso Puig; Octubre 2003. http://www.mundobot.com/tecnica/svi2c/spsvi2c.htm

- [14] Announced Specification of HS-311 Standard Servo; Hitec RCD Korea Inc. http://www. hitecrcd.com/Servos/HS311.pdf

9. BIOGRAFÍAS

Patricio J. Cruz Dávalos
Nacido el 8 de octubre de 1980, cursó su instrucción primaria en la Escuela Municipal "Eugenio Espejo". Sus estudios secundarios los realizó en el Colegio Municipal "Sebastián de Benalcázar". Se graduó en la Escuela Politécnica Nacional como Ingeniero en Electrónica y Control en enero de 2005. Realizó una pasantía en la Unidad de Comunicaciones de la Gerencia de Oleoducto de Petroecuador. Fue Ingeniero Desarrollador en el Área de I+D de la empresa Autotrack, Cia. Ltda. Trabajó como técnico electrónico en el área de prensa de la gerencia de producción del Grupo "El Comercio". Actualmente es Asistente de Cátedra I de la carrera de Ingeniería en Electrónica y Control en la Escuela Politécnica Nacional.

e-mail: patoc20@hotmail.com

Cristóbal R. Villegas Santillán
Nacido el 8 de enero de 1981, cursó su instrucción primaria en la Escuela San José "La Salle". Sus estudios secundarios los realizó en el Colegio San Luis "Gonzaga". Se graduó en la Escuela Politécnica Nacional como Ingeniero en Electrónica y Control en enero de 2005. Actualmente trabaja en el Departamento de Instrumentación y Control de Procesos de REPSOL YPF ECUADOR.

Nelson G. Sotomayor
Nació en Quito el 9 de Septiembre de 1971. Realizó sus estudios secundarios en el Instituto Nacional Mejía. Se graduó en la Escuela Politécnica Nacional como Ingeniero en Electrónica y Control en 1999. Obtuvo su título de Magíster en Ingeniería industrial en junio del 2006. Actualmente desempeña el cargo de Profesor Principal T/C en el Departamento de Automatización y Control Industrial de la Escuela Politécnica Nacional. Además es miembro de la subcomisión académica permanente de la Carrera de Ingeniería en Electrónica y Control.
Áreas de interés: robótica móvil, informática y redes, microcontroladores, automatización y control industrial

e-mail: nsotomayor@hotmail.com